SE Assignment 3

Homework Submission Guidelines

1. **Due date: 28.06.20 at 23:55**
2. **הגשה בזוגות בלבד!**
3. הקוד חייב להיכתב בהתאם למוסכמות כתיבת הקוד בקורס כולל תיעוד כנדרש. קוד שלא עומד בדרישות יגרור הורדת ניקוד
4. ההגשה מתבצעת ב- Moodle באזור המיועד על ידי אחד מהשותפים בלבד
5. כל איחור בהגשה יגרור הורדת 20% מהציון בכל יום
6. פורמט הגשת התרגיל נמצא בקובץ ההנחיות ב-Moodle . כל חריגה מפורמט זה תגרור ציון 0
7. שאלות והבהרות ייכתבו **רק** בפורום ייעודי שייפתח לתרגיל הבית ב-Moodle

מטרת התרגיל:

- עבודה עם מספר מחלקות
- כתיבת מתודות ומשתני מחלקה
- תרגול כימוס, רב צורתיות, ירושה, מחלקות אבסטרקטיות וממשקים
- עבודה מול מאגרי מידע (קריאה, שליפת נתונים והצגתם)
- Generics
- תבניות עיצוב

נושא התרגיל:

בניית אינדקס מסמכים ואחזור מסמכים ביחס לשאילתות בוליאניות.

תיאור מאגר המידע:

בתרגיל זה נבנה אינדקס ומנוע אחזור עבור שאילתות בוליאניות.
מאגר המידע כולל 1500 מסמכים הנמצאים בתיקיה "AP_Coll_Parsed". כל קובץ מכיל מסמך אחד בפורמט שיתואר בהמשך.
המסמכים הנמצאים בתיקייה הינם כתבות חדשותיות אשר נלקחו מהמאגר AP.
הקובץ BooleanQueries.txt מכיל 5 שאילתות בוליאניות עבורן נצטרך לאחזר את כל המסמכים אשר עונים על התנאי הבוליאני בשאילתות אלו.

תיאור המערכת:

1. **מסמכי הטקסט**

```
<DOC>
<DOCNO> AP880212-0001 </DOCNO>
<TEXT>
More than 150 former officers of the
overthrown South Vietnamese government have been released from a
reeducation camp after 13 years of detention the official Vietnam
News Agency reported Saturday
The report from Hanoi monitored in Bangkok did not give
specific figures but said those freed Friday included an
exCabinet minister a deputy minister 10 generals 115
fieldgrade officers and 25 chaplains
….
</TEXT>
</DOC>
```

דגשים:

1. כל קובץ טקסט של מסמך מתחיל ומסתיים בתגיות: <DOC> , </DOC>
2. שם המסמך הינו מזהה ייחודי וממוקם בין התגיות <DOCNO> , </DOCNO>
3. טקסט המסמך, אותו נצטרך לאנדקס ממוקם תמיד בין התגיות <TEXT> , </TEXT>
4. **שימו לב** – עשויים להיות מספר בלוקים של טקסט במסמך כך שכל בלוק טקסט עטוף בין תגית <TEXT> ל-</TEXT>.
5. כל התגיות מופיעות תמיד בשורה נפרדת מלבד צמד התגיות DOCNO שעוטפות ביניהן את שם המסמך.
6. כל סימני הפיסוק הוסרו מהמסמכים.
7. הטקסט עשוי להכיל אותיות גדולות וקטנות.

2. **שאילתות בוליאניות נתונות לפי תחביר Reverse Polish Notation:**

```
Southwest Airlines OR
southwest Airlines OR Africa NOT
Winner
death cancer OR US NOT
Liberty US AND labor oil AND NOT
```

דגשים:

1. שאילתה בוליאנית מורכבת ממילים ומהאופרטורים הלוגיים הבאים: AND, OR, NOT.
2. שימו לב שמילות השאילתה עשויות להופיע עם אותיות גדולות וקטנות.
3. השאילתות הבוליאניות כתובות במבנה שנקרא: Reverse Polish Notation. מומלץ להיכנס ללינק הבא שיעזור לכם במימוש שלב אחזור המסמכים ביחס לשאילתה:

https://www.programcreek.com/2012/12/leetcode-evaluate-reverse-polish-notation/

4. ניקח לדוגמא את השאילתה הבאה:

   southwest Airlines OR Africa NOT

   בשאילתה זו, נבקש לאחזר את כל המסמכים המכילים את המילה southwest או את המילה Airlines אבל לא מכילים את המילה Africa.

עבור שאילתות בסגנון של Winner נאחזר פשוט את כל המסמכים המכילים מילה זו.

<u>ה-API איתו נשתמש בתרגיל זה + דוגמאות שימוש</u>

1. קובץ Utils.java מכיל 4 מתודות סטטיות אשר עשויות לשמש אתכם בפתרון תרגיל זה. מומלץ לעבור על התיעוד של מתודות אלו טרם תחילת פתרון התרגיל.
2. קובץ DocumentRetrieval.java מכיל את המתודה הראשית של התוכנית. אין לשנות קובץ זה. המימוש הנתון יעזור לכם במימוש המחלקות והחלקים החסרים בתוכנית.
   מתודת ה-main תקבל שני קלטים מהמשתמש:
   1. הנתיב לתיקייה AP_Coll_Parsed אשר מכילה את המסמכים.
   2. הנתיב לקובץ השאילתות הבוליאניות BooleanQueries.txt.

<u>בניית האינדקס – Inverted Index:</u>
בתרגיל זה עליכם להשתמש בתבנית הנתונים **Factory Method** עבור יצירת סוגי אינדקסים.
בתרגיל זה ניצור שני סוגי אינדקסים שונים אשר כל אחד יאנדקס את המסמכים בצורה אחרת:

1. אינדקס בו יש חשיבות לאות גדולה/קטנה - CaseSensitive
2. אינדקס אשר מאנדקס את המילים כאשר כל מילה תהיה כתובה באותיות קטנות – CaseInsensitive
   a. ליצירת מילה המכילה רק אותיות קטנות יש להשתמש בפקודה: toLowerCase() ששייכת למחלקה String.

מבנה האינדקס הינו מבנה נתונים ייחודי אשר ממפה כל מילה לקבוצת המסמכים המכילה אותה. לכן מבנה נתונים זה מקבל את השם invertedIndex, כלומר האינדקס ההופכי (ה-key הינו מילה וה- value הינו קבוצת מסמכים).
לדוגמא:

```
‘the’ -> (AP880219-0002, AP880314-0254)
‘sanctions’ -> (AP880221-0077, AP880314-0254)
‘african’ -> (AP880222-0029)
```

כלומר, המילה “the” מופיעה במסמכים AP880219-0002 ו- AP880314-0254.
שימו לב:

1. באינדקס שהינו CaseSensitive יכול להיות גם key עבור “The”. באינדקס שהוא CaseInsensitive יכול להופיע רק key אחד – “the” אשר יכיל את כל המסמכים שמכילים אותו. תזכורת שעבור אינדקס זה נדאג שכל המילים מכל המסמכים יופיעו עם אותיות קטנות.
2. את קבוצת המסמכים המכילה את המילה ב-key יש לשמור ממוינים לפי השם שלהם.

3. מכל סוג של אינדקס, נרצה ליצור אובייקט אחד ויחיד ולכן נשתמש בתבנית העיצוב **Singleton**.
4. בעת יצירת אינדקס מסוג CaseSensitive תודפס למסך ההודעה הבאה:
“New CaseSensitive index is created”

במידה וכבר נוצר אובייקט לאינדקס זה, ברגע שננסה ליצור אובייקט נוסף תודפס למסך ההודעה הבאה:
“You already have a CaseSensitive index”
והאובייקט הקיים יוחזר.

5. בעת יצירת אינדקס מסוג CaseInsensitive תודפס למסך ההודעה הבאה:
“New CaseInsensitive index is created”

במידה וכבר נוצר אובייקט לאינדקס זה, ברגע שננסה ליצור אובייקט נוסף תודפס למסך ההודעה הבאה:
“You already have a CaseInsensitive index”
והאובייקט הקיים יוחזר.

דגשים כלליים:

1. היכן שניתן יש להשתמש במחלקות אבסטרקטיות/ממשקים.
2. יש להשתמש בהרשאות הגישה המתאימות למתודות ולשדות פנימיים בהתאם לעקרון הכימוס.
3. היכן שניתן יש לכתוב מתודות בצורה גנרית.
4. בטיפול השאילתות הבוליאניות באמצעות Reverse Polish Notation ניתן להשתמש במחלקה Stack הקיימת כבר ב-java.

הנחות:

1. ניתן להניח שמילות השאילתה מופיעות בלפחות מסמך אחד.
2. עבור אחזור מסמכים מהאינדקס שהוא CaseInsensitive יש לדאוג שמילות השאילתה יהיו כולן עם אותיות קטנות. עבור אינדקס שהינו CaseSensitive כמובן שיהיה הבדל בין מילת שאילתה שהופיעה עם אותיות גדולות לבין אותה מילה שהופיעה רק עם אותיות קטנות.
3. שם המסמך הינו String.

הכנות טרם תחילת התרגיל:

1. הורדת קובץ ה-zip של תרגיל זה המכיל את כלל הקבצים הנדרשים:
   a. Utils.java.
   b. DocumentRetrieval.java.
   c. AP_Coll_Parsed – תיקיית המסמכים שיש לאנדקס.
   d. BooleanQueries.txt – קובץ השאילתות הבוליאניות.
   e. OutExample.txt – קובץ הפלט שאמור להתקבל עבור הרצה תקינה של המחלקה DocumentRetrieval עבור קלט המסמכים המופיעים ב-AP_Coll_Parsed ורשימת השאילתות המופיעה בקובץ BooleanQueries.txt.
2. פתיחת פרויקט חדש ב- Intellij והוספת **קבצי ה-java** תחת התיקייה src.

הוראות כלליות:

1. יש לקרוא את כלל ההנחיות וההסברים **לאט ומספר פעמים** טרם תחילת העבודה.
2. לצורך פתרון התרגיל מומלץ לחזור על התרגולים וההרצאות וכן להיעזר באינטרנט.
3. יש לבדוק שהקוד מתקמפל ללא שגיאות.
4. **בדקו שאין לכם שכפול קוד**. יש להעביר מתודות/משתנים משותפים למחלקות אב.
5. **אין לשנות כלל** את הקובץ Utils.java ואת DocumentRetrieval.java.
6. עליכם לתעד את הקוד כנדרש באנגלית בלבד, יש לתעד לפי סגנון התיעוד המופיע במודל.
7. הקוד חייב להיכתב על פי מוסכמות כתיבת הקוד בקורס. קוד שלא יעמוד במוסכמות לא יזכה במלוא הניקוד.

הוראות הגשה

1. יש למלא אחר הוראות ההגשה בהתאם לקובץ הדרישות ״הנחיות כלליות לפתרון והגשת תרגילי הבית״ בmoodle.
2. יש להגיש את **כלל** קבצי הקוד הקיימים ואלו שהוספתם תחת תיקיית src ללא הוספת תיקיות נוספות.
3. הגשה אלקטרונית **בלבד** דרך אתר הקורס ב-moodel
4. אין להגיש אותו הקובץ פעמיים, התרגיל יוגש על ידי **אחד** מבני הזוג.
5. תרגיל בית שלא יוגש על פי הוראות ההגשה – **לא ייבדק**
6. יש להקפיד על יושרת הכנת התרגיל וההגשה
7. קוד אשר לא יעבור קומפילציה – **ציון 0**